

# **Understanding Assurance Cases: An Educational Presentation in Five Parts**

## **Module 1: Foundation**

C. Michael Holloway  
c.michael.holloway@nasa.gov

Senior Research Computer Engineer  
Safety-Critical Avionics Systems Branch  
NASA Langley Research Center, Hampton, Virginia, U.S.A.

# **UNDERSTANDING ASSURANCE CASES**

## *MODULE 1: FOUNDATION*

**C. MICHAEL HOLLOWAY**

NASA LANGLEY RESEARCH CENTER  
C.MICHAEL.HOLLOWAY@NASA.GOV

*People generally quarrel because they cannot argue. - Gilbert K. Chesterton*

**VERSION 2.1**

2020-07-14

This material was originally created in 2015-16, as part of the Explicate '78 project. The project was supported in substantial part by the Assurance Case Applicability to Digital Systems task under the reimbursable interagency agreement with the Federal Aviation Administration for Design, Verification, and Validation of Advanced Digital Airborne Systems Technology (IAI-1073 Annex 2 for NASA; DTFAC-10-X0008, Modification 0004 for the FAA). The original presentations were delivered to a selected group of FAA civil servants and NASA Langley personnel. The audio was recorded and partial transcripts (containing only the words spoken by the presenter, Mr. Holloway) produced. The intent from the beginning was to collect the material into a form that could be made available publicly. The text adheres closely to the original transcript, except where changes have been made to the original presentation since it was first given, as part of work for for NASA IA-303333/FAA IA NO 692M15-19-T-00029 Annex 1/TO 1. The full collection consists of six documents, which are available electronically through <https://shemesh.larc.nasa.gov/arg/uac.html>.

Hello everybody.

Welcome to the first module in an educational series about Understanding Assurance Cases. In this module, we will examine the **Foundation** of the assurance case concept.

Because talking about the foundations will involve talking quite a bit about *argument*, the quotation you see here from Gilbert K. Chesterton is particularly appropriate:

“People generally quarrel because they cannot argue.”

[Chesterton, G. K. 2002. *The Collected Works of G.K. Chesterton*. Electronic edition: (v35) Illustrated London News, 1929-1931. Charlottesville, Va: InteLex Corporation.]

When we talk about argument we will *not* be talking about emotion-filled disagreements; instead, we’ll be talking about rational, careful discussion of reasons for thinking one thing rather than another.

My hope is that this hour will be interactive. There will be several times when I’ll ask you a question, and many times when I’ll stop to give you a chance to ask me questions. Nevertheless, feel free to interrupt me at *any* point if you have a burning question that you can’t hold until later. I’ll do my best to extinguish it.

[Question to participants: Does anyone have any questions or comments that you want to make now at the beginning?]

Before going any further, I feel duty-bound to alert you to an intentional act of deception underlying this, and the other, modules.

Within the assurance case community, intramural debates abound about a variety of topics we will discuss. Except in rare instances the existence of these debates<sup>1</sup> is intentionally ignored or mentioned only briefly in this material. Here’s why.

Disagreements exist about terms, definitions, notations, philosophy, procedures, tools, and just about everything else.

The depth of the disagreements ranges all the way from *shallow* differences in preferences (which term best denotes a particular concept, for example), to rather *deep* philosophical differences (the feasibility and desirability of formalizing assurance arguments, for example).

Spending *too much* time on these disagreements would likely make this material deeply confusing; but spending *too little* time on them might hinder your understanding of some materials you may come across.

---

<sup>1</sup> By using the word ‘debates’, I’m intentionally obscuring something else, too, namely the fact that *some* of the disagreements have all of the attributes of quarrels (not *all* by any means, but *some*).

In trying to strike a balance, what I've chosen to do is *not* highlight the areas of disagreement on the slides (except occasionally where it seems essential), but to mention the disagreements where appropriate in my words accompanying the slides.

[Question to participants: Any questions about this issue?]

One other quick note before we proceed: All images you see were either created by me (Michael Holloway) or are in the public domain via CCo 1.0 Universal.

## LEARNING OBJECTIVES

A person completing Module 1 should be able to

- ❖ Define assurance case
- ❖ Explain the key concepts of assurance cases and recognize various terms for them
- ❖ Identify some existing notations for expressing assurance cases
- ❖ Enumerate characteristics that an assurance case should have

*People generally quarrel because they cannot argue. - Gilbert K. Chesterton*

Let's discuss learning objectives.

By the time we're finished today, I hope that you'll be able to do at least four things.

First, provide a definition for the term 'assurance case'. Although I've not listed it on the slide, I also expect that you'll be able to provide definitions for more specific variants such as 'safety case' or 'security case'.

Second, explain the key concepts of assurance cases and recognize various terms used for those key concepts. This is one area where the slides and my oral commentary will both note some differences within the community.

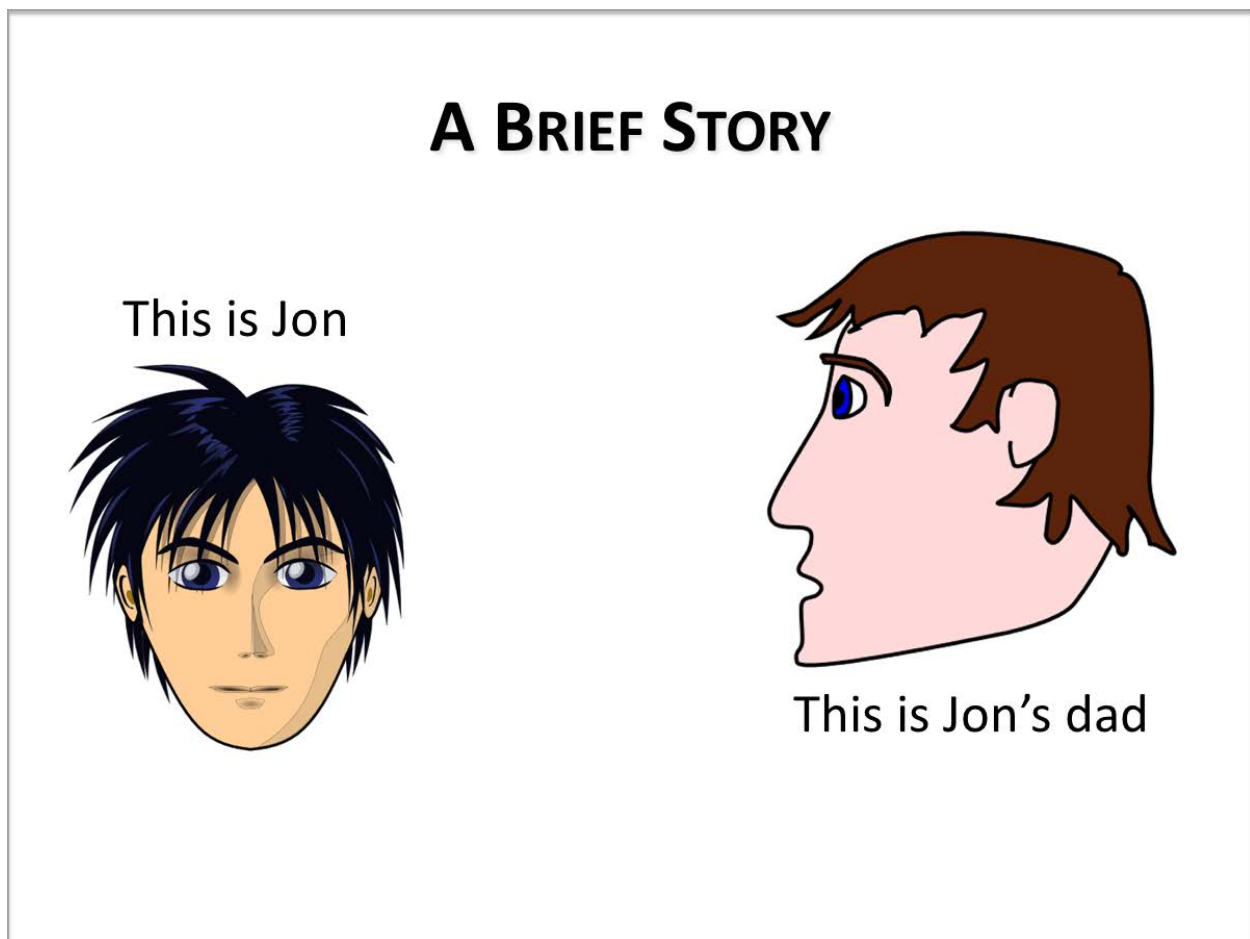
You should also be able to identify some existing notations for expressing assurance cases.

Finally, and perhaps most importantly, you should be able to enumerate characteristics that an assurance case should have. By this, I do *not* mean to be able to list characteristics of a *good* assurance case versus a *bad* assurance case, but simply to be able to list characteristics to distinguish between something that *can be legitimately* called an assurance case, and something that *does not* deserve the name.

In other words, if someone comes to you with a document that they claim is an assurance case, you should be able to read over the document and say, “Yes, it is an assurance case”, or “No, it is not an assurance case.”

Material about how to distinguish between a good case and a bad case will be covered in Module 3 about Evaluation.

[Question for participants: Any questions about these learning objectives?]



The majority opinion among educators today is that telling stories is a very good thing; so here's a story for you.

The young fellow with black hair on the left is called Jon.

The older brown-haired fellow on the right is Jon's dad. His name is Mike. (Note: the original presentation included automated slides with the images of Jon and Mike speaking the appropriate dialog. Reproducing those slides here is unnecessary.)

One day, Jon comes up to his dad, and says:

“Tim will give me a ride to the game.”

Because Jon’s dad knows nothing about Tim’s driving ability, he asks,

“Is he a safe driver?”

“Yes, he is,” replies Jon.

Not willing to simply trust Jon’s rather information-free assertion, Jon’s dad asks,

“How do you know?”

Jon, perhaps because he’s a tad miffed that his Dad didn’t just accept without question his claim that Tim is a good driver, replies,

“It’s just one of those things I know.”

Jon’s dad, undoubtedly a bit *more* than a tad miffed with this response, tells Jon,

“That’s not good enough. Try again.”

Jon thinks for a little while, and then says,

“No one says he’s **not** a safe driver.”

Jon’s dad, wondering how big the ‘no one’ set is, asks Jon,

“How many people have you asked?”

Jon’s reply is a bit disappointing, but not particularly surprising to his dad,

“Um, well, one, but ... ” followed by a pause, which eventually ends with Jon continuing,

“He passed the state test to get a license, so he must drive safely.”

Jon’s dad pauses before replying, deciding whether to ask Jon how in the world he thought that one person was enough to attest to Tim’s driving ability. After a second or two, he decides to let it pass, and instead address the license issue,

“Just being legal doesn’t mean he’s safe.”

At this point, Jon recognizes that he’s totally lost control of the conversation, and asks, exasperatingly,

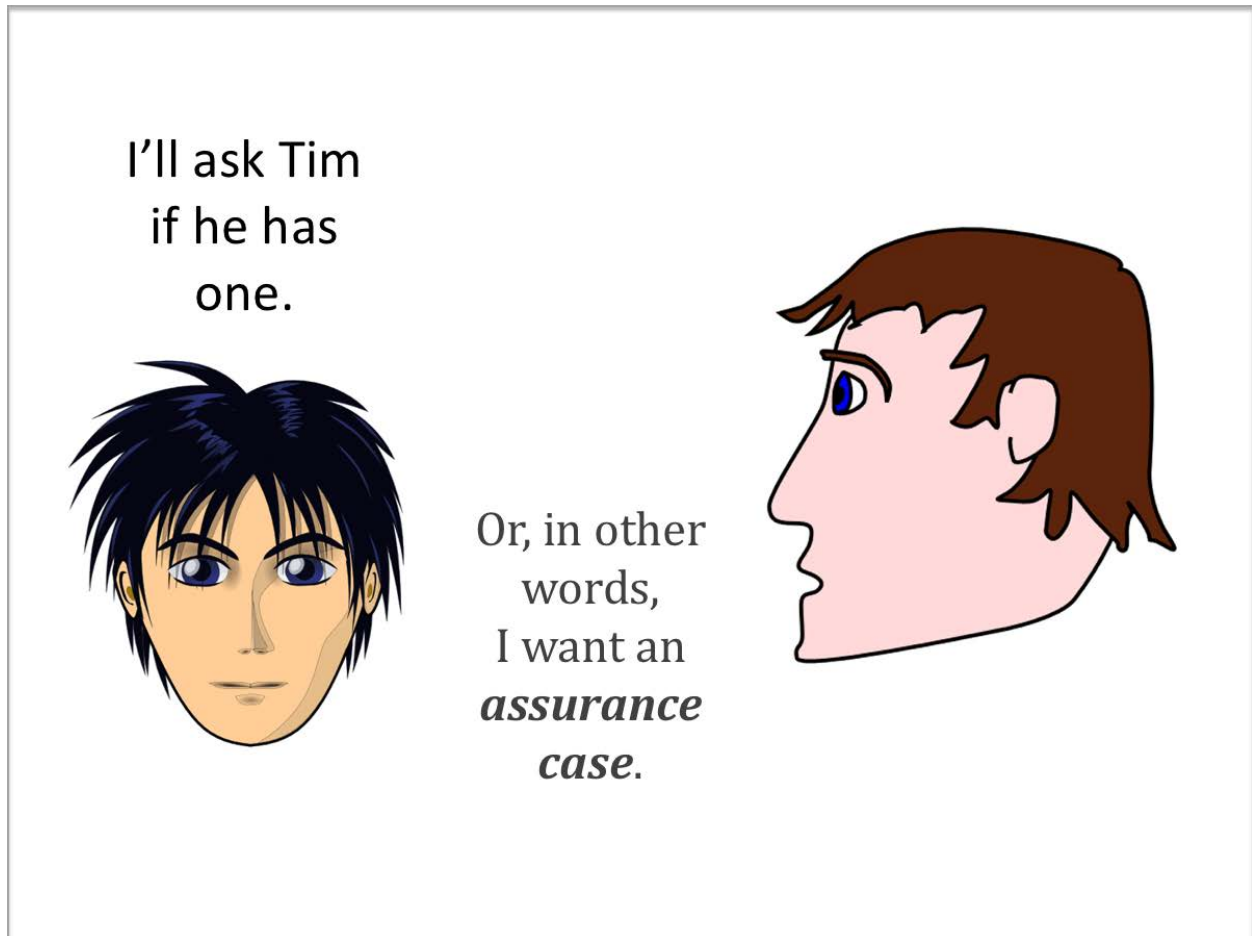
“Well, Dad, what’s gonna convince you to let me ride with Tim?”

Jon’s dad decides to ignore Jon’s not-entirely respectful tone, and simply says,

“Reasons ... *Good* reasons to believe Tim will get you there and back in one piece.”

And after a brief pause, he adds, “Or, in other words, I want an *assurance case*.”

After an even longer pause, Jon responds, “I’ll ask Tim if he has one.”



And thus ends our story, leaving us to wonder: Does Tim have an assurance case? Upon seeing it, will Jon’s dad let him ride with Tim? Who will win the game? Will Jon get home safely? And, what about Naomi?

If you don’t understand that last question, type it into your favorite search engine either before or after the phrase “love of chair.”

Of course this story is a bit silly, and not entirely realistic, but it does illustrate indirectly, and I hope memorably, many of the basic concepts that we’re going to discuss directly now.

So, what is an assurance case?

# A WORKING DEFINITION

An assurance case is  
an explicit argument  
that a system or service  
is acceptable for its intended use.

Avoids defects suffered by currently popular definitions

Here is a working definition: *An assurance case is an explicit argument that a system or service is acceptable for its intended use.*

I call this a ‘working definition’ because it is the definition that we will use throughout this series of educational modules; it captures, I believe, all of the essential elements that distinguish an assurance case from something else<sup>2</sup>.

This definition does not exactly match any specific definition currently used commonly in the literature or existing standards and guidelines.

Those definitions, in my opinion, suffer from various defects, and this definition is designed to avoid those defects.

The most common definitions one sees in the literature are definitions derived from early definitions of ‘safety case’. They include in the definition notions of ‘goodness’ that I don’t think appropriately belong, beginning, for example with something like this: “A reasoned and compelling argument ...”

---

<sup>2</sup> The phrase ‘assurance case’ (or ‘safety case’) is used in a variety of ways, not all of which require the existence of an explicit argument. These other uses are not relevant for the purposes of this module. The interested reader can explore the following paper, which was written after these materials were originally created: Graydon, P. J. 2017. “The Safety Argumentation Schools of Thought.” *3rd International Workshop on Argument for Agreement and Assurance (AAA)*. November 13-15, 2017. Tokyo, Japan. Accessed October 11, 2018. <http://hdl.handle.net/2060/20180000378>.

Well, ‘reasoned’ and ‘compelling’ are certainly characteristics one wants in a *good* assurance case, but including them in the basic definition is akin to defining ‘student’ as something like, “an enthusiastic and diligent learner ....” What then do you call folks running about our schools and colleges who are not particularly diligent and perhaps a tad bored?

The definitions currently used in some standards and guidelines also are encumbered with ‘goodness’ ideas, while additionally suffering from verbosity and poor wording choices.

For example, ISO/IEC 15026-1, section 3.1.3 gives this ugly definition of ‘assurance case’: “reasoned, auditable artifact created that supports the contention that its top-level claim (or set of claims), is satisfied, including systematic argumentation and its underlying evidence and explicit assumptions that support the claim(s).” There are also 5 lines of ‘note’ attached to the definition that specify the contents of an assurance case. [ISO/IEC 15026-1:2013, Systems and software engineering - Systems and software assurance - Part 1: Concepts and vocabulary, Switzerland, Geneva.]

The simple definition that you see here does not suffer from any of these defects, so it is the one we’re going to use.

Most of the rest of this presentation will involve discussing further the three main aspects of the definition.

## A WORKING DEFINITION

An assurance case is  
an <sup>2</sup>explicit argument  
that a <sup>1</sup>system or service  
is <sup>3</sup>acceptable for its intended use.

Avoids defects suffered by currently popular definitions



The first aspect we'll discuss, quite briefly in fact using only a single slide, is 'system or service'.

The majority of our time will be spent discussing the second aspect, which is 'explicit argument.' While we're discussing explicit argument, we'll touch on the third aspect: 'acceptable for its intended use.'

We'll then talk a little bit more specifically about this aspect. As you probably have already surmised, One can easily change 'acceptable' to 'safe' to give a definition for safety case, and 'acceptable' to 'secure' to give a definition for security case.

[Question for participants: Does anyone have any questions at this point?]

Before we proceed, I should probably mention that the word in this working definition that would generate the most controversy in a room full of assurance case people is almost certainly the first 'is'. Some people would strongly insist that 'is' *must* be replaced by 'contains'. Their insistence stems from wanting to emphasize that any practical assurance case will need to have additional information besides just the bare argument itself. Items such as a system description, a list of system hazards, and a discussion of the safety management system are among the sorts of information they have in mind. Although I agree that such items usually need to be part of the case, I also believe that they can also legitimately considered to be part of the argument; hence, 'is' is appropriate, and 'contains' unnecessary.

## **'SYSTEM OR SERVICE'**

- ❖ Emphasis is not on distinguishing between the two but on acknowledging applicability to more than just engineered artifacts
  - The assurance case asked for by the dad would concern Tim's driving ability
- ❖ Think of some aviation systems or services

The use of the phrase ‘system or service’ in the definition is not intended to emphasis distinguishing between the two, but rather to make clear that assurance cases do *not* just apply to engineered artifacts (which is what many people often think of when they hear the word ‘system’). Assurance cases at least as well (and in fact, as we’ll see in Module 2 are perhaps most firmly established) for operational procedures, maintenance activities, collections of ‘best practices’, and other such things that are often not called ‘systems’ but rather ‘services’.

As a specific example, motivated by the tragic, apparently intentional crash in France recently<sup>3</sup>, one can imagine an assurance case being developed to cover new rules and procedures for when and how to lock the cockpit door.

I’m sure that all of you can think of many aviation systems or services. And I suspect that an assurance case could be developed for any of them, so long as it is possible to identify what the system or service is intended to provide.

That’s all that I intend to say about ‘system or service’, unless someone has a question.

[Question for participants: Are there any questions about ‘system or service’?]

We’ll move now to talking about **explicit argument**.

I’ve structured this discussion based on the terminology and ideas described by Stephen Toulmin in his seminal book *The Uses of Argument*. [S. E. Toulmin. 1958. *The Uses of Argument*. Cambridge, UK: Cambridge University Press, updated ed. 2003.], with some changes in terminology I’ve made over the years.

There are plenty of other ways the discussion could be structured, but Toulmin’s ideas have strongly influenced the notion of assurance cases over the years, and I think his general framework is easier to understand than most others, and corresponds more clearly to many people’s intuitive notions; and in those places where his ideas may not be quite so clear, I’ve made some modifications I hope will add clarity.

Not everyone agrees with my opinion. Formalists, for example, tend not to like Toulmin’s ideas very much, and it is safe to say that he was not overly fond of their ideas, either.

The next several slides build on one another. Almost certainly you’ll have questions after seeing one slide, but it is likely that at least some of your questions may be answered on the next slide, or perhaps a couple down the road; for this reason, I’d like for you to hold your questions for a little bit, until I explicitly ask for them. Also, please keep in mind that we’ll first be talking about simple, self-contained arguments, which rarely exist in pure form in the real world. Later on, we’ll talk a bit about real world arguments, which are usually rather messy, and quite complicated.

In the following slide, you see a simple form of an explicit argument.

It has three parts: a **conclusion**, one or more **premises**, and **reasoning**.

---

<sup>3</sup> At the time of the original presentation, the Germanwings flight 9525 crash investigation was not complete, but the evidence was becoming compelling that the co-pilot caused it intentionally.

The arguer (let's say it is you) wants someone else (let's say it is me) to believe that a particular **conclusion** is true.

To convince me, you'll give me some **premises**, which are statements that you are confident that I will accept as true, and explain your **reasoning** why the truth of the **premises** is sufficient to justify the truth of the **conclusion**.

## EXPLICIT ARGUMENT

State a **Conclusion** that you believe is justified  
by applying **Reasoning** to **Premise(s)**

Friendly  
Argument  
Notation

Believing  
**conclusion**  
is justified by applying  
**reasoning**  
to these premises  
**premise1**  
**premise2 ...**  
**premiseP**

Or, to put this into what I call the Friendly Argument Notation (FAN), you might write Believing **conclusion** is justified by applying **reasoning** to these **premises**.

The **conclusion** is *what* you want me to believe.

The **premises** are things that you think I *already* believe.

The **reasoning** explains *why* taking the step from believing the **premises** to believing the **conclusion** is an 'appropriate and legitimate one' (to use Toulmin's language).

Here are three examples ...

Given (**premise**) "*Annette was born in Lynchburg, Virginia*" you should believe (**conclusion**) "*Annette is a US citizen*" because (**reasoning**) "*People born in Virginia are US citizens.*"

Why is the fact that Annette was born in a city in Virginia enough to justify belief that she is a US citizen? The **reasoning** provides the answer to the question.

Second example.

Given (**premise 1**) “ $A = B$ ” and (**premise 2**) “ $B = C$ ” you should believe (**conclusion**) “ $A = C$ ” because of (**reasoning**) “*the transitive property of equality.*”

Why is  $A=B$  and  $B=C$  enough to justify believing  $A=C$ ? The transitive property of equality explains.

And finally, one more example, In which I’ll use a different ordering of elements, you should believe “*Angela is eligible to run for President of the US,*” given these three **premises**: “*Angela is a natural born US citizen,*” “*Angela is 54 years old,*” and “*Angela has lived in the US all her life.*”

Why? Because of this **reasoning**: “*The eligibility requirements of Article II Section 1 of the Constitution are natural born citizen, at least 35 years old, and having lived in the US for 14 years.*”

I’m sure you can think of many examples of your own.

You may also be able to think of different names that you’ve heard given to the three parts: **conclusion**, **premises**, and **reasoning**.

Toulmin himself tended to refer to **premises** as ‘data’, the **conclusion** as a ‘claim’, and **reasoning** as the ‘warrant’.

Other terms used for concepts similar to **reasoning** include simply ‘reasons’, quite confusingly the word ‘argument’ itself, and (as just noted) ‘warrant’.

We’ll talk more about alternate terms a bit later, and I’ll mention my reasons for preferring **premise**, **conclusion**, and **reasoning**.

[Question for participants: Does anyone have any questions about what these three terms mean?]

So far I’ve concentrated on the ‘argument’ part of ‘explicit argument.’

Real life, however, is full of arguments in which at least one of the parts is implicit rather than explicit. We can see a simple example by returning to our story.

Recall that Jon told his dad Tim had passed the state test to get a license.

## IMPLICIT → EXPLICIT

He passed the state test to get a license, so he must drive safely.



Believing

**Tim drives safely**

is justified by applying

**The implicit belief that only safe drivers pass the test**

to these premises

**Tim passed the drivers license test**

In the context of the story, Jon's statement about Tim passing the state test, can be seen as part of an implicit argument.

The **conclusion**, which Jon wants his dad to believe, is that "*Tim drives safely.*"

In this snippet of the conversation, Jon gives a **premise** for this **conclusion**, namely "*Tim passed the test.*"

He is implicitly applying the **reasoning** "*Only safe drivers pass the test*" to the "*Tim passed the test*" **premise** to justify belief in the **conclusion**.

Jon's dad is not swayed by this argument, because he (correctly) does not accept the implicit **reasoning**.

That's an example of the **reasoning** being implicit, which is a situation that is quite common, so common in fact that many approaches to teaching argumentation do not directly address the concept of **reasoning** directly at all, but rather fold it into their discussion of **premises**.

We could also give examples in which one or more of the **premises** is implicit, or even in which the **conclusion** is implicit, or at least not stated specifically. All of these situations of implicitness seem to be fairly common in certain aspects of engineering practice today.

One of the major distinguishing factors of an assurance case approach is the explicit statement of a top level **conclusion**. The explicit statement of a **conclusion** to be established makes it possible for an assurance case to articulate an argument with that same **conclusion**.


As you already may be thinking, the very simple form of argument that we've seen so far may be a bit too simple. You are right.

Often the **conclusion** may need to be expressed in less than absolute terms with **qualification**. That is, the arguer may not be asserting that the **conclusion** is necessarily always and certainly true given the **premises** and the **reasoning**.

Returning to the story, perhaps Jon's implicit argument is really more something like what is shown here on the slide.

## EXAMPLE OF QUALIFICATION

He passed the state test to get a license, so he must drive safely.



Believing  
*It is highly likely Tim drives safely*  
is justified by applying  
*The implicit belief that unsafe drivers often fail the test*  
to these premises  
*Tim passed the drivers license test*

Given that Tim passed the drivers' license test (the **premise**), and (implicitly) knowing (the **reasoning**) that unsafe drivers often fail the test, then it is highly likely (the **qualification**) (but not necessarily certain) that Tim drives safely (the **conclusion**).

Or here's another example that is a bit more technical. People without knowledge about how software is approved for use on civil aircraft will have to take my word that the example is realistic.

# ANOTHER EXAMPLE

## Believing

*(to a level of confidence that meets airworthiness requirements)* the software will not contribute to a failure of system function resulting in a catastrophic failure condition for the aircraft

is justified by applying

The FAA accepts DO-178C for assessing software

to these premises

The data items for the software show compliance with all DO-178C Level A objectives

For level A software on a civil aircraft, one of the **conclusions** that we want to be able to believe (here's the **qualification**) "*to a level of confidence that meets airworthiness requirements*" is that "*The software will not contribute to a failure of system function resulting in a catastrophic failure condition for the aircraft.*"

Speaking in fairly simple terms, a primary **premise** that is often used to justify this *conclusion* can be said to be "*The data items for the software show compliance with all DO-178C Level A objectives.*"

Why does this **premise** justify the **conclusion**?

Because of the **reasoning**: "*The FAA accepts DO-178C for assessing software.*"

I realize that this example oversimplifies reality a bit, so please don't dissect it too much at this point; it is just intended as an illustrative example of how **conclusions** may need **qualification**.

It also may prompt some of you to think that this model of argument may still be incomplete. Once again, you are correct.

One of Toulmin's insights was the recognition that sometimes it is not possible to state the **reasoning** in such a way as to encapsulate all that's necessary to explain why the **reasoning** justifies accepting the **conclusion** based on the **premises**. Something additional may be needed.



I'll explain this something additional by quoting Toulmin, making minor changes to match our slightly different terminology.

“In defending a **conclusion**, we may produce our **premises**, our **reasoning**, and the relevant **qualification** and yet find that we have still not satisfied our challenger; for he may be dubious not only about this particular argument but about the more general question whether the **reasoning** is acceptable at all.”

“Presuming the general acceptability of this **reasoning** (he may allow) our argument would no doubt be impeccable.... But does not that **reasoning** in its turn rest on something else?”

“Standing behind our **reasoning** there will normally be other assurances, without which the **reasoning** themselves would possess neither authority nor currency. These other things we may refer to as the **backing** of the **reasoning**.”

We're going to use the term **backing**, too, although we're going to ignore some details and distinctions that Toulmin makes in his book. (Recall that I said that our argument discussion was based on Toulmin's ideas, not that it would be identical to them.)

For our purposes, you can think of **backing** as explaining why the **reasoning** applies or, if you prefer a slightly different wording, reasons for accepting the **reasoning**.

## BACKING ADDED TO EXAMPLE

### Believing

*(to a level of confidence that meets airworthiness requirements)* the software will not contribute to a failure of system function resulting in a catastrophic failure condition for the aircraft

### is justified by applying

The FAA's acceptance of DO-178C for assessing software (because DO-178C was developed and approved by international experts and believed by them to be adequate)

### to these premises

The data items for the software show compliance with all DO-178C Level A objectives

(other ways to do this are also possible)



Here is a statement of **backing** added to the argument you just saw. It asserts that we can accept the **reasoning** on account of the fact that “*DO-178C was developed and approved by international experts and believed by them to be adequate.*”

Again, this is just an example. I think it is a fairly realistic example, but I’ve not been as careful in the wording as would be necessary to turn this into something more than just an example. Of course, there are other (I tend to think, better) ways to incorporate backing an argument, but we’ll leave discussing them until another day.

We’re almost done with the framework, but not quite. There’s one more element of argument that we need to mention.

But before we do that, I’ll pause to give you a chance to ask questions.

The final element of argument we will discuss is the notion of **defeaters**, which deals with circumstances in which the general authority of the **reasons** to justify the **conclusion** must be set aside.

An example or two should help make the concept clear.

Think back to the simple example argument I gave earlier about Angela being eligible to run for President. It had **premises** about her place of birth, her age, and the length of her residency in the US, and the **reasons** referred to the eligibility requirements established in the US Constitution.

A **defeater** is “*Angela has already been elected twice to the office of President.*”

In such a case, Section 1 of the 22<sup>nd</sup> amendment makes her ineligible, despite her meeting the standard eligibility requirements; the **reasoning** that usually justifies the **conclusion** does not do so in this special case.

We can also expand the DO-178 example to include a possible **defeater**.

# DEFEATER ADDED TO EXAMPLE

Believing

*(to a level of confidence that meets airworthiness requirements)* the software will not contribute to a failure of system function resulting in a catastrophic failure condition for the aircraft

is justified by applying

The FAA's acceptance of DO-178C for assessing software  
(because DO-178C was developed and approved by international experts and believed by them to be adequate)

to these premises

The data items for the software show compliance ...

unless

The requirements to which the software was developed specify some unsafe behaviors

Because DO-178's guidance is based on the assumption (which we haven't stated in the argument) that the system safety process has created requirements that, if satisfied, will ensure safety, the argument that we've given so far is also based on that (unstated) assumption.

Thus, if, for a particular instance of software, "*The requirements to which the software was developed specify some unsafe behaviors*", then the argument no longer holds water: it does not establish the truth of the **conclusion**. It has been defeated.

One more slide, and then I'll pause again for questions.

## KEY TERMS – OTHER NAMES

<b>Premise</b>	evidence, <del>solution</del> , data, assumption
<b>Conclusion</b>	claim, goal, thesis
<b>Reasoning</b>	warrant, (premise), argument <small>(unfortunately)</small> , <del>strategy</del>
<b>Defeater</b>	rebuttal, counter-argument, counter-evidence
<b>Backing</b>	reasoning, justification, argument <small>(unfortunately)</small>
<b>Qualification</b>	level-of-confidence, likelihood
<b>Backing</b>	(context)

So far, I've specifically introduced six concepts that make up an argument: **premise**, **conclusion**, **reasoning**, **defeater**, **qualification**, and **backing**.

As we've talked about these concepts, I've mentioned some of the other names used for the concepts; this slide lists the most popular alternative terms.

Within the assurance case community, the most common terms tend to be *evidence* (instead of **premise**), *claim* or *goal* (instead of **conclusion**) and (confusingly) *argument* (instead of **reasoning**).

One of the two most popular notations for expressing assurance cases is called claims-arguments-evidence (or CAE), where *claims* are pretty much the same as **conclusions**, *arguments* are very similar to **reasoning** (or perhaps **reasoning** plus **backing**), and *evidence* is equivalent to certain types of **premises** (more on that later).

The other most popular notation (the Goal-Structuring Notation – GSN) uses the terms *goal*, *strategy*, and *solution*, with *goal* being pretty much equivalent to **conclusion**, *solution* being generally equivalent to certain types of **premises**, and *strategy* serving a role somewhat analogous to **reasoning** and **backing**, though not exactly like it.

The OMG's Structured Assurance Case Metamodel S-A-C-M talks about *claims*, *arguments*, *evidence*, and *reasoning*, among other terms.

I personally think the community is not best served by some of these choices of terminology. Particularly unfortunate in my opinion is the overloading of the term

*argument* to refer not only to the overall argument, but also to that **part** of the argument that links **premises** with **conclusions**. It is much clearer and less confusing to use **reasoning** (and, if needed, **backing**) for that part.

**Conclusion** is a better term than *goal* or *claim* in my opinion because it does not carry the potentially negative connotations that can be associated with those terms. *Claim*, in particular, tends to suggest to some people, myself included, something that is *asserted* to be true, but in reality is most likely *not* true. Consider, for example, the sentence, “My daughter claimed she did her homework last night.” Do you think the daughter did her homework?

In saying that I’m not fond of the common terminology, I’m not saying that there are no legitimate reasons that this terminology was chosen and continues to be used; there are reasons (based on analogies to some other disciplines, for example), which are deemed more than adequate, by plenty of folks, so perhaps I’ve made a bigger deal out of this than I should, but I don’t think so.

You see here at the bottom of the slide two additional terms that are often important in practice important but which are not explicitly part of the Toulmin-based argument model: **bindings** (which is the term I prefer) and the somewhat analogous GSN term **context**. For now all you need to do is remember that these terms exist. I’m not going to talk about these anymore in this module.

We’ll talk about all these terms quite a bit more in the future, particularly in the Evaluation and Creation modules.

Right now, however, I want to stop to take questions, of which I’m sure that are several.

[Question for participants: What are your questions?]

We’ve looked at the elements that make up arguments. Now we need to talk a little bit about types of arguments.

For our purposes, arguments can be grouped into two categories: *deductive* arguments and *inductive* arguments.

# TWO GROUPINGS OF ARGUMENTS

## ❖ Deductive argument

- Warrant concerns form & may be **valid** or **invalid**
- A deductive argument with valid warrant and true premises is called a **sound** argument.
- A sound argument **guarantees** a true conclusion

## ❖ Inductive argument

In a deductive argument, the **reasoning** is about the form of the argument. It can be either *valid* or *invalid*.

If a deductive argument has valid **reasoning**, and true **premises**, it is called a *sound* argument.

A sound argument *guarantees* a true **conclusion**.

Or in other words, it is *not possible* for a deductive argument to have valid **reasoning**, true **premises**, and a false **conclusion**.

Here are two examples of deductive arguments.

## EXAMPLE DEDUCTIVE ARGUMENTS

Believing

$$A = C$$

is justified by applying

Transitive property of equality

to these premises

$$A = B ; B = C$$

Believing

No FAA employees are overworked

is justified by applying

classical logic EAE-1 syllogism

to these premises

No civil servants are overworked

All FAA employees are civil servants

The first one I mentioned early on in our discussion about the elements of argument. It is a simple instantiation of the transitive property of equality.

The second example is new. It asserts the following: Given (**premise 1**) “No civil servants are overworked” and (**premise 2**) “All FAA employees are civil servants” we should believe the **conclusion** that “No FAA employees are overworked” because the form (**reasoning**) is a “EAE-1 syllogism” from classical logic, which is known to be one of the valid forms of syllogisms.

The first of these examples is a sound deductive argument: the form is valid, and (so long as we’re talking about mathematical equality) the **premises** are true; hence, the **conclusion** is necessarily also true.

The second example is a valid deductive argument, but it is not sound (and hence the **conclusion** not necessarily true), because one of the **premises** (“No civil servants are overworked”) is false.

Please note, and remember always, that just because an unsound argument is given with a particular **conclusion**, does not mean that the **conclusion** is necessarily false.

A sound argument guarantees a true **conclusion**; but an unsound argument by itself tells us *nothing* about the truth of the **conclusion**. It may be false. It may be true (just badly argued for). We do not know.

[Question for participants: Any questions about deductive arguments before I talk a bit about the other main type of argument?]

## TWO GROUPINGS OF ARGUMENTS

### ❖ Deductive argument

### ❖ Inductive argument

- Not to be confused with mathematical induction (which is really a species of deductive argument)
- Reasoning assessed according to **strength**
- **Strong** reasoning and true premises increase confidence that the conclusion is true
- **Weak** reasoning or false premises should have no effect on confidence

The first thing that everyone needs to remember about inductive arguments is that they are *not* related to mathematical induction, which is really a species of deductive argument.

The terms valid / invalid, sound / unsound don't really apply to inductive arguments, 'though you will hear those terms used by some folks.

It is much better to talk in terms of *strength* when it comes to inductive arguments.

An inductive argument with strong **reasoning** and true **premises** should increase confidence that the **conclusion** is really true; whereas weak **reasoning** or false **premises** should (by themselves) have no effect on confidence.

Here are two simple examples of inductive arguments.

# EXAMPLE INDUCTIVE ARGUMENTS

Believing

I will not die on my next flight

*strong*

is justified by applying

Vast majority of deaths on flights are due to accidents  
to these premises

My next flight is on a US carrier

US carriers rarely have fatal accidents

Believing

The software has no bugs

is justified by applying

Testing tends to uncover bugs

to these premises

A test plan has been developed

The test plan has been executed

*weak*

Given “My next flight is on a US carrier” and “US carriers rarely have fatal accidents”, I believe “I will not die on my next flight”, because the “Vast majority of deaths on flights are due to accidents.”

Of course, as most of you may be thinking, if I was following the Toulmin-based framework more closely, I should include a **qualification** in the **conclusion**, but I’ve left it out for simplicity, and to enable me to make a point in just a minute.

In the second example, we are arguing that given (**premise 1**) “A test plan was been developed” and (**premise 2**) “The test plan has been executed”, we should believe (**conclusion**) “The software has no bugs” based on the **reasoning** that “Testing tends to uncover bugs”.

[Question for participants: What do you think about the strength of these two arguments?]

I’m inclined to say that the first argument is fairly strong, while the second argument is pretty weak.

The first argument could be made even stronger by qualifying the **conclusion**, into something like “It is very unlikely that I will die on my next flight.”

This illustrates an important point about inductive arguments: the strength of the argument depends on *all* parts of it, not just on (for example) the **reasoning** or the **premises**.



The second example argument as it stands is quite weak, since (among other things) a tendency to uncover bugs does not imply that *all* existing bugs are uncovered. Even if we qualified the **conclusion** a bit, the argument is still going to be rather weak, since (among other things) the **premises** tell us nothing about the quality of the test plan.

Please always remember that a weak inductive argument does not necessarily mean that the **conclusion** is false. It simply means that *this particular argument* ought not give you confidence that its **conclusion** is true. Perhaps there is a strong argument with the same **conclusion** but different other constituent parts.

If, on the other hand, there exists a strong argument with an opposite or contradictory **conclusion**, then that new argument should provide confidence in the falsity of the original **conclusion**.

Similarly, a poor assurance case does not necessarily mean that the system or service is *not* acceptable for its intended use; but it may well indicate some problems.

[Question for participants: Does anyone have questions before we continue?]

So far, we've been talking about arguments mostly in the context of examples contrived to illustrate particular ideas. What about the real world?

## ARGUMENTS IN THE WILD ...

- ❖ Are usually rather complicated
  - **Premises** for the initial argument are themselves **conclusions** of additional arguments with **premises** that are **conclusions** of still more arguments and so on to quite a depth
- ❖ Rarely state explicitly all the **premises** or provide complete **reasoning**
- ❖ Never consist of only deductive arguments
- ❖ May be very difficult to evaluate
  - Module 3 will address this issue in more detail

Well, arguments in the wild tend to be quite different from the simple examples we've seen in at least four ways.

First, real arguments are usually rather complicated. In particular, **Premises** for the initial argument are themselves **conclusions** of additional arguments with **premises** that are **conclusions** of still more arguments and so on to quite a depth. The argument in any real assurance case for why its top level **conclusion** should be accepted will certainly take such a form. The **premises** for the top level **conclusion** will almost certainly not be obvious truths, but rather statements that will need to be supported by argument themselves.

Eventually the assurance case should stop with sub-arguments with **premises** whose truth can be agreed upon by all relevant parties. A purported 'assurance case' that isn't grounded in such **premises** doesn't deserve to be called an 'assurance case'.

Second, real arguments rarely state explicitly all of the **premises** or provide complete **reasoning**.

Combatting this tendency to leave many things unstated is one of the goals of the assurance case approach.

An assurance case, to be worthy of the name, needs to have sufficiently explicit information (or at least references to information contained elsewhere) to enable evaluators and users of the case to know the intended meaning of all aspects of the argument.

Third, real arguments almost never consist of only deductive arguments.

As I mentioned earlier, this is an area about which there is some controversy. No one disputes that it is true that current assurance cases inevitably contain some inductive arguments. The disputes center around whether there may be advantages to be gained from making deductive as many arguments as possible; or perhaps by using a normalized structure that isolates inductive arguments into specific parts of the overall argument.

Those who believe that there are advantages to be gained point to (among other things) the simpler evaluation of deductive arguments (much of which could likely be automated).

Those who believe otherwise point to (among other things) the inherent non-formality of many relevant concepts, the likelihood of a huge increase in argument size with a related decrease in human readability, and a skepticism that the sorts of problems solved by formalism are actual problems in real assurance cases.

And finally, as a consequence of these three characteristics, real arguments in the wild may be very difficult to evaluate. Hence the need for a separate module in this series talking about evaluating assurance cases.

We're getting near the end, but this is a good place to stop to ask for questions.

I'm not going to spend a lot of time talking about notations, but I do want to let you know that there are various ones.

# SOME ARGUMENT NOTATIONS

## ❖ Graphical

- Toulmin diagrams – not intended as a notation
- Goal Structuring Notation (GSN)
- Claims-Arguments-Evidence (CAE) ...

## ❖ Textual

- Friendly Argument Notation (FAN)
- Regular or structured prose
- Argument outline
- Tables ...

In an earlier version of this model, I used Toulmin diagrams fairly extensively, but I stopped that practice because Toulmin never intended his diagrams to be used that way. Instead I introduced the textual Friendly Argument Notation (FAN).

I've mentioned GSN and CAE, which are the two most common graphical notations used for assurance cases. The website <http://www.goalstructuringnotation.info/> is a good place to visit if you want more information about GSN. To explore CAE further point your favorite browser at <https://www.adelard.com/asce/choosing-asce/cae.html> There are other graphical notations, also, as you may imagine.

There are also or textual ways of representing arguments besides FAN, ranging from unstructured prose, through structured prose, outlines, and tables.

Also, about ten years ago I wrote a conference paper repeating the same example using several notations. It is available at <http://hdl.handle.net/2060/20080042416>. [Holloway, C. M. 2008. "Safety Case Notations: Alternatives for the Non-Graphically Inclined?" *IET 3rd International Conference on System Safety*. 21-23 October 2008, Birmingham, UK.]

That's all that I plan to say about notations, but will be happy to field questions if you have any.

# A WORKING DEFINITION

An assurance case is  
an explicit argument<sup>✓</sup>  
that a system or service<sup>✓</sup>  
is acceptable for its intended use.<sup>3</sup>

Returning to our working definition, we've covered two of the three main parts, and while talking about argument we've alluded to what needs to be said about the third part: 'acceptable for its intended use'.

There's only one more thing that I want to say about it.

The top level **conclusion** is where 'acceptable for intended use' is going to be mainly defined; thus a good choice of top level **conclusion** is critical for the success of an assurance case approach. Some critics of the assurance case approach have seemingly missed this point, leveling much of their attacks on badly worded top level **conclusions** as if somehow the approach itself requires people to start with bad ones.

We'll be talking about how to *recognize* a good top level **conclusion** in some detail in module 3 about assurance case evaluation; and about how to choose a good top level **conclusion** in some detail in module 4 about assurance case creation.

We are almost done with module 1.

Before we quit, however, I want you to think a bit about how you would complete a sentence that begins, "It isn't an assurance case if ...."

We've covered the foundations of assurance cases in sufficient detail that I think you can complete this sentence with several characteristics that distinguish an assurance case from something else.

Here are four things that I think are appropriate completions.

## IT ISN'T AN ASSURANCE CASE IF ...

- ❖ It does not state a top level conclusion
- ❖ It does not articulate an argument for the conclusion
- ❖ It is not grounded in premises whose 'truth' can be agreed on by all relevant parties
- ❖ It contains too little information to define the meaning of all aspects of the argument

It isn't an assurance case if ... It does not state a top level **conclusion**. If someone claims to have an assurance case but you can't find the **conclusion** the case is supporting, then you're justified in telling them, "This is not an assurance case."

It isn't an assurance case if ... It does not articulate an argument for the **conclusion**. If someone claims to have an assurance case but there's no argument, then you're justified in telling them, "This is not an assurance case."

It isn't an assurance case if ... It is not grounded in **premises** whose 'truth' can be agreed upon by all relevant parties. If someone claims to have an assurance case but it ends with **premises** whose truth is no more certain than that of higher level conclusions then you're justified in telling them, "This is not an assurance case."

Within the assurance case community, these 'grounded **premises**' tend to be called *evidence*, for a variety of reasons, including analogies to common usage from other fields, and a general belief that there is value in having a specific term for the concept.

So most of my colleagues within the community would probably say something like this: "It isn't an assurance case if ... it is not grounded in evidence." If you prefer that formulation from what I've written here, then you'll be in good company.

Finally, it isn't an assurance case if ... It contains too little information to define the meaning of all aspects of the argument. If someone claims to have an assurance case but it leaves terms or concepts undefined, then you're justified in telling them, "This is not an assurance case."

At the beginning, I listed four things that I hoped you'd be able to do by the end of this module.

Here are those four things recast in the form of questions.

## REVIEW OF LEARNING OBJECTIVES

Are you able to

- ❖ Define assurance case?
- ❖ Explain the key concepts of assurance cases and recognize various terms for them?
- ❖ Identify some existing notations for expressing assurance cases?
- ❖ Enumerate characteristics that an assurance case should have?

*People generally quarrel because they cannot argue. - Gilbert K. Chesterton*

Think to yourself how you'd answer these questions. If you are not confident in your answers, consider reviewing the materials again.

If you have questions or comments about this material, contact its author at `c.michael.holloway@nasa.gov`.